

2-D VISUAL SERVOING WITH INTEGRATION OF MULTIPLE PREDICTIONS OF MOVEMENT BASED ON KALMAN FILTER

Jorge Pomares Baeza
Fernando Torres Medina
Pablo Gil Vázquez

*Automatics, Robotics and Computer Vision Group (<http://www.disc.ua.es/gava>)
Physics, Systems Engineering and Signal Theory Department
University of Alicante
Crtra. San Vicente, s/n, 03080 Alicante*

Abstract: In this paper, a visual servoing for the tracking of objects in a plane using a fixed camera system is shown. The method proposed of tracking makes use of a variable number of trackers determined automatically. The integration of the predictions of each of them affords a flexible and robust system of tracking. The studies of efficiency carried out reflect that the delays produced are suitable for use in visual servoing. The proposed system can be generalized to tracking three-dimensional objects with movable cameras. *Copyright © 2002 IFAC*

Keywords: Tracking, Visual motion, Automatic control (closed-loop), Delay analysis, Estimation algorithms, Image processing, Prediction methods.

1. INTRODUCTION

Visual servoing consists of using vision in the lowest level of the control hierarchy. The great benefit that can be obtained from using visual servoing is the accuracy that is obtained with a configuration in closed loop, which makes the system relatively insensitive to calibration errors. A general idea of visual control systems can be observed in (Hutchinson *et al.*, 1996; Corke 1996). Two types of visual servoing are distinguished: one based on position and the other based on image. In the latter, the control uses coordinates of the features caught in the image plane. However, in the position-based visual servoing, the control is made in the cartesian space. In this article, the approach of image based visual servoing is used.

The processes of tracking and control are the main components of a visual servoing. Real time tracking is the object of research (Nickels and Hutchinson, 2001; Isard and Blake, 1998) which motivates the appearance of new algorithms oriented to the processing of sequences of images at high speed. The development of these algorithms, together with the recent improvement in processing times of the computers, affords a better performance of the systems of visual control that use them as well.

Visual servoing requires that the techniques used for tracking be applicable to real time. Often, valid tracking systems are obtained by using marks or knowledge of the objects tracked, obtained beforehand (Papanikolopoulos and Khosla, 1993). Nevertheless, a compromise between the speed of process and flexibility is needed to find, which allows us to make the tracking in non-structured environments, even with deficiencies in some component of the visual servoing. This article shows a robust and flexible tracking system in real time for application in visual servoing. A tracking system based on the Kalman filter has been investigated, which automatically adapts the amount of information to be processed, depending on the goodness of the tracking. As such, if the filter tracks the object suitably, the amount of information about each image to be processed by the filter will be diminishing and, on the contrary, if imperfections are detected, an adjustment is made in the tracking, thus increasing the processing time.

The delays produced in these systems influence the global performance, as shown in (Sharkey and Murray, 1996). One way of reducing the latency times in the visual feedback is to simplify the computer vision systems. However, these delays will always be present and must be considered to be able

to improve the performance of the visual servoing (Barreto *et al.*, 2000). Unlike what happens in other cases (Oh and Allen, 2001), in the proposal of this paper is used a fixed-camera system and, therefore, does not require any control of the camera's position to do the tracking.

The rest of the article is organized as follows: In section 2, is shown the architecture of the visual servoing used. In sections 3 and 4, the main components of the visual control system are described. That is to say, the tracking algorithm, the control system and the results obtained are discussed. Finally, some of the conclusions that can be extracted from this research are presented, along with suggestions for possible future studies.

2. ARCHITECTURE OF THE SYSTEM

Figure 1 shows, schematically, the architecture used.

The system has a flat plane, around which two electric cars circulate. Car A, which is called 'the chased car', is handled by a user and is the permanent reference for Car B, 'the chaser', which is controlled so that it stays at a fixed distance from the chased one. The chased car is driven by an external source of voltage (V_1). The chaser is controlled by the Control PC so that the chase adjusts to the imposed specifications.

The vision system is made up of a fixed camera, CCD SONY XC75CE, with optics of 7,5mm and a Matrox Corona image card lodged in the Control PC, in which there is also an input/output card (Analogic/Digital), model CIO-DAS 1600 (ComputerBoards) used to provide voltage to the chaser car (V_2). The tests have been done on a Control PC with a Pentium III processor of 800 MHz and 128 Mbytes of RAM.

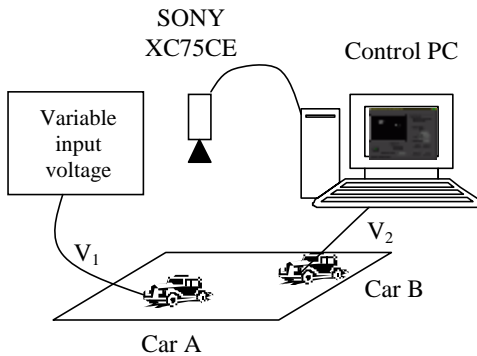


Fig. 1. System architecture.

3. TRACKING

In this section, the tracking method is described with special emphasis on the real-time integration of different predictions of movement, for the generation of just one prediction which is more robust and reliable.

3.1 Matching of characteristic zones

A movable object is in position (x, y) at moment t , and in position $(x+\lambda, y+\mu)$ at moment $t+1$, where λ and μ are the displacements that the object has undergone in each direction, considering that the tracking is being done on a plane. The objective of this algorithm is to determine the value of these displacements by making a comparison in the vicinity of (x, y) to determine the new location of the moving body.

A window, v , with a size of $n \times n$ pixels, located at the center of gravity of a given car is used. The coordinates of these pixels are represented with the set W . The window will have intensities associated to each of its boxes, which will be equal to the intensities of the pixels on which they are located (the value of the intensity of a pixel with coordinates $[x, y]^T$ at moment t is equal to $I([x, y]^T; t)$). This window is also called the reference window and is the region of the image that it is trying to follow in the sequence. This paper assumes that the image stored in the reference window is either flat or that the effects of depth and/or perspective are negligible. This flat pattern, v , will undergo a transformation at moment t , symbolised by a homogenous matrix A_t , on its original position at moment t_0 ($t > t_0$). In order to determine the new position of v , an A_t will be selected that minimizes the following objective function:

$$F(A_t) = \sum_{(x,y) \in W} \left(I(A_t[x, y]^T; t) - I([x, y]^T; t_0) \right)^2 \cdot w(x, y) \quad (1)$$

Where $w(x, y)$ is a weight function.

In the following paragraphs is determined the position of the reference window at moment $t+\tau$, knowing that the displacement undergone at moment t was A_t obtained in (1). For the calculation of the new position, the transformation undergone, $A_{t+\tau}$, is determined in a way that is analogous to the one used in (1). A distance function that provides the error made in the comparison is defined. This function of distance will be the objective function to be minimized:

$$F(A_{t+\tau}) = \sum_{(x,y) \in W} \left(I((A_t A') \cdot [x, y]^T; t + p) - I([x, y]^T; t_0) \right)^2 w(x, y) \quad (2)$$

Where A' represents the transformation between moments t and $t+\tau$, and $A_{t+\tau} = A_t A'$.

Let us now suppose that all the transformation matrices, A , are divided into sub-matrices R (rotation) and T (transfer) and linearized about the point $(R', T') = 0$ as shown in (Hager and Toyama, 1998). The objective function to be minimized is the following:

$$F'(R', T') = \sum_{(x,y) \in W} \left(\nabla I([x, y]_t) \cdot (R'[x, y] + T') + (J([x, y]_t) - I([x, y]_{t_0})) \right)^2 w(x, y) \quad (3)$$

Where:

$$J([x, y]_t) = I(R_t[x, y] + T_t; t + t) \quad (4)$$

The next problem to be considered is in what places of the image the comparison will be made. That is to say, the environment that maximizes the probability of finding the object that is being tracked.

3.2 Integration of multiple predictions

The technique “matching of characteristic zones” uses only one piece of information about vicinity, i.e., only one reference window, since the position of a point $(x+\lambda, y+\mu)$ at moment $t+1$ is determined from the position (x, y) at moment t .

To improve the algorithm, the integration of several reference windows (which in the following is also denominated “seeds”) is used to afford a robust algorithm. These seeds are concentrated randomly on the vehicle whose location is trying to find, and each one of them verifies its location in the following frame. Therefore, each seed will have a reference window, v , associated to it (see figure 2). A set of n reference windows is maintained, and the number n has a variability, as will be determined later.

To demonstrate how this algorithm works, in the following is supposed the use of n estimations of movement, each one made by using the “matching characteristic zones” technique described in Section 3.1. It is also supposed that “estimation” is the position in which the seeds are found, separately, at the following moment $t+1$, and “prediction” is the place where the centre of the tracked car is predicted to be at the following moment $(t+2)$.

Given n movement estimations, each of them will determine that the undergone displacement has been (x_i, y_i) units. The average of all these estimations will be \bar{x} and the variance will be σ_x^2 .

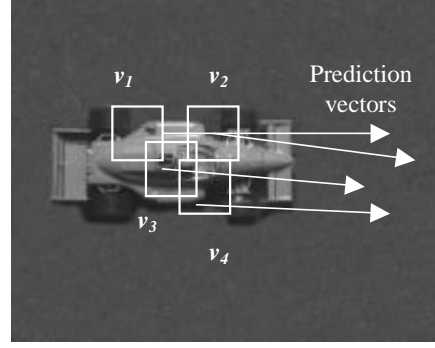


Fig. 2. Extended figure of the tracking in which the reference window of each seed is shown together with the prediction provided by each one.

The same consideration will hold for the coordinate y .

The real value that is considered for the centre of gravity predicted by the recursive estimator will be modified by a measurement error:

$$\begin{aligned} x_{mi}(n) &= x_i(n) + r_{xi}(n) \\ y_{mi}(n) &= y_i(n) + r_{yi}(n) \end{aligned} \quad (5)$$

Where n is the sample of time in which the value of the corresponding variable is obtained. In the implementation, the measurement error represented in the equation (5) is a Gaussian curve with an average of 0 and the measurement error has an associated variance which is empirically measured. The next consideration is to determine how the prediction is obtained. Let us suppose that, with the use of the filter, the displacement of the center of gravity has been $\hat{x}_i(n), \hat{y}_i(n)$. If the center of gravity at a previous moment was $x_{mi}(n-1), y_{mi}(n-1)$, the position of the new center of gravity will be:

$$\begin{aligned} x_{mi}(n) &= x_{mi}(n-1) + \hat{x}_i(n) \\ y_{mi}(n) &= y_{mi}(n-1) + \hat{y}_i(n) \end{aligned} \quad (6)$$

Therefore, in the following is determined the prediction of the displacement of the centre of gravity at moment $t+2$ from the estimations between moments t and $t+1$. To do so, a recursive estimator is used, which takes the following form:

$$\begin{aligned} \hat{x}_i(n) &= \hat{x}_i(n-1) + K_i(n) \cdot (x_{mi}(n) - \hat{x}_i(n-1)) \\ \hat{y}_i(n) &= \hat{y}_i(n-1) + K_i(n) \cdot (y_{mi}(n) - \hat{y}_i(n-1)) \end{aligned} \quad (7)$$

Where $K_i(n)$ is the gain of the filter.

In order to correctly specify the gain term of the equation (7) some new suppositions from the

observation of the random variables presented in the model is made. The research supposes that $r_{xi}(n)$ and $r_{yi}(n)$, and that the predictions of each seed (with variances $S_{pi}^2(n)$) are independent variables. With these considerations, the error of the variance of the present prediction, obtained from the error of the variances of the previous measures, will be:

$$\hat{S}_i^2(n|n-1) = \hat{S}_i^2(n-1|n-1) + S_{pi}^2(n)$$

The value $S_{ni}^2(n)$ is a parameter of noise produced in the calculation of the estimation and is used for the subsequent calculation of the gain. This factor is a fixed value that has been adjusted empirically. With this consideration, the optimal gain will be the following expression:

$$K_i(n) = \frac{\hat{S}_i^2(n|n-1)}{\hat{S}_i^2(n|n-1) + S_{ni}^2(n)} \quad (9)$$

The variance of the result of the estimation is given by:

$$\hat{S}_i^2(n|n) = \hat{S}_i^2(n|n-1) - \frac{\hat{S}_i^2(n|n-1)}{\hat{S}_i^2(n|n-1) + S_{ni}^2(n)} \quad (10)$$

The variance represented in (10), is used in the following iteration of the algorithm to determine the optimal value of the gain. This gain is also used to determine the number of seeds needed. Empirical thresholds have been defined so that the greater the variance is, the greater the number of seeds used to improve the tracking will be.

4. CONTROL SCHEME

Figure 3 shows the structure used for the control system. The input, X_d , is shown in this scheme, which indicates the desired distance for the tracking.

$F(z)$ is the transfer function of the plant. It has been obtained empirically with continuous techniques in the time domain and its value in the Laplace domain is the following:

$$F(s) = \frac{1.025}{(1 + 67s)(1 + 98.53s)(1 + 74.64s)} \quad (11)$$

The delays in any visual servoing can make that the operating of the system and the control is worse and more difficult. The delays causes that the margins of stability of the system are reduced. In the described system, the delays do not have a fixed value and vary with factors as it can be the number of seeds used in a given moment. Nevertheless, an average delay value obtained empirically has been considered for the system analysis.

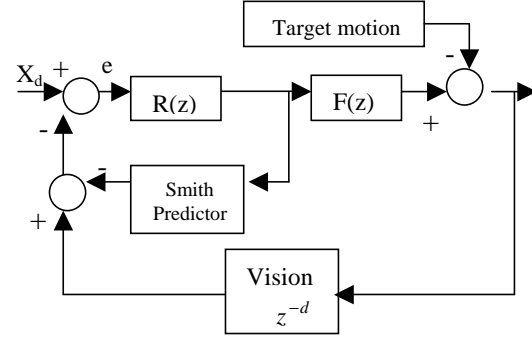


Fig 3. Block diagram of the visual servoing

In the scheme of figure 3, the delays produced by the computer vision system used are shown. Furthermore, the model of the system is well-known, so that a Smith predictor (Aström K.J and Wittenmark B., 1997) can be used, as shown in (Sharkey P., Murray D., 1993) in order to eliminate the effects produced by the delays. An average delay d , equal to 10 milliseconds for the image processing, has been determined. The Smith predictor will have the general form of:

$$F(z) - F(z)z^{-d} \quad (12)$$

The Smith predictor tends to eliminate the non desired dynamic effects which have been produced by the delays, generating the future values of the feedback. The use of this prediction along with the system regulation produces in the practice results, a better stability of the visual servoing system.

The scheme of figure 3 shows the use of a compensator, $R(z)$, in the control loop. A PID compensation has been used (the controller is designed for the following specifications of the system: settling time $t_s=500$ msec, a maximum overshoot of 5% and a steady-state error less than 10%) whose value in the Laplace domain appears in the equation (13). Figure 4 shows the improvement in the transient response of the system, with the use of the PID compensator presented in (13).

$$R(s) = \frac{s + 0.01015}{s + 0.005329} \cdot \frac{s + 0.0010472}{s + 0.0000562} \quad (13)$$

In figure 4, the transient response of the system with and without using a PID controller is presented. In this figure is shown that the transient response of the system without the PID controller has more oscillations. The overshoot and the settling time of the regulated system is reduced considerably with the introduced controller action, this way is obtained a better global performance of the visual servoing.

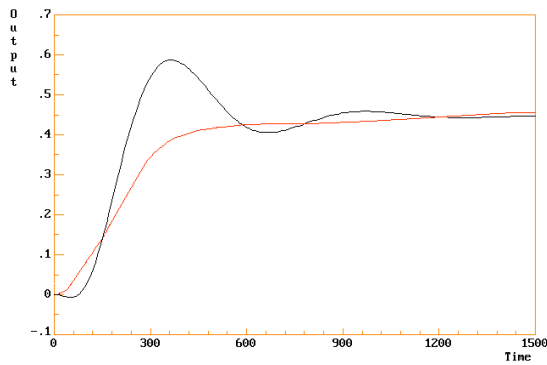


Fig. 4. Original and regulated system (PID)

The tests carried out have shown a correct tracking, even in circumstances such as defects in the illumination or temporary occlusions. Unlike the use of just one prediction (Pomares J., *et al.*, 2001), the use and integration of multiple predictions allows a more flexible and robust tracking with variations and non-structured environments.

5. CONCLUSIONS

In this paper is proposed the use of a tracking algorithm based on the Kalman filter, for visual servoing. In such systems, the delays are critical and special attention has therefore been paid to this matter in the article. The compromise between speed of processing and precision in detection has been arrived at by determining the number of seeds in the tracking algorithm.

An automatic method for the determination of the number of seeds is used. An overall study of the tracking system is made, integrating it into the global scheme of the control system.

The proposed tracking algorithm is sufficiently robust to variations of the environment, noise and defects of illumination, as demonstrated in the tests carried out. Methods for the control of the trajectories of the moving objects are developed in order to avoid external factors and occlusions. In the research are implemented them so that in case of the non-detection of the movement, due to an occlusion, the position of the object is inferred statistically. The use of the above-mentioned techniques allows tracking in non-structured environments.

We are presently working on the generalization of the algorithm to tracking objects in 3-D, as well as the use of movable cameras to extend the vision rank during the tracking.

REFERENCES

- Aström K. J., B. Wittenmark (1997), *Computer-Controlled Systems-theory and design*, Pp. 215 – 217. Prentice Hall
- Barreto J.P., J. Batista, H. Araujo. (2000). Model Predictive Control to Improve Visual Control of Motion: Applications in Active Tracking of Moving Targets. *Proc. of the International Conference on Pattern Recognition (ICPR '00)*.
- Corke P. I. (1996). *Visual controls of robots. High-performance visual servoing*. Research studies press Ltd.
- Hager G. D and K. Toyama. (1998). X Vision: A Portable Substrate for Real-Time Vision Applications. *Computer Vision and Image Understanding*. **69**, No. 1. 23-37
- Hutchinson S., G. D. Hager, P. I. Corke. (1996). A Tutorial on Visual Servo Control. *IEEE Transactions on Robotics and Automation*. **12** No 5. 651-670.
- Isard, M., A. Blake. (1998) CONDENSATION: Conditional Density Propagation for Visual Tracking. *Int. Journal of Computer Vision*. **29(1)**. 5 – 28.
- Nickels K., S. Hutchinson. (2001). Model-Based Tracking of Complex Articulated Objects *IEEE Transaction on Robotics And Automation*. **17** No 1, 28 – 36
- Oh P.Y., P. K. Allen. (2001) Visual Servoing by Partitioning Degrees of Freedom. *IEEE Transactions on Robotics and Automation* **17**, No 1. 1 – 17.
- Papanikolopoulos N. P. and P. K. Khosla. (1993) Adaptive robotic visual tracking: Theory and experiments *IEEE Transaction on Automatic control*. **38**. 429-445.
- Pomares J., P. Gil, F. Torres. (2001) “Control visual para el seguimiento de móviles en un plano”. *Proc. Jornadas de Automática 2001*.
- Sharkey P.M. and D. W. Murray. (1996) Delay versus performance of visually guided systems. *IEE Proc. Control Theory Appl.*, **143(5)**. 436-447.
- Sharkey P. and D. Murray. (1993) Coping with delays for real-time gaze control. *SPIE. Sensor Fusion VI*. **2059**, pp 292-304.